UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/722,408	11/28/2003	Subhashini Subramaniam	SUN-007/030215	7400

51121 7590 07/07/2010

LAW FIRM OF NAREN THAPPETA
C/o Landon-IP Inc.,
1725 Jamieson Avenue
Alexandria, VA 22314

EXAMINER	
CHOI, PETER H	
ART UNIT	PAPER NUMBER
3623	

NOTIFICATION DATE	DELIVERY MODE
07/07/2010	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

lfnt2000@yahoo.com
oracle@iphorizons.com
intercomm@iphorizons.com

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Application Number: 10/722,408
Filing Date: November 28, 2003
Appellant(s): SUBRAMANIAM, SUBHASHINI

Narendra Reddy Thappeta (Reg. No. 41,416)
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed March 29, 2010 appealing from the Office action mailed October 29, 2009.

(1) Real Party in Interest

The examiner has no comment on the statement, or lack of statement, identifying by name the real party in interest in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The following is a list of claims that are rejected and pending in the application: claims 1, 3-6, 8-11, 13-16, 18-21, 23-26, 28-31 and 33-36.

(4) Status of Amendments After Final

The examiner has no comment on the appellant's statement of the status of amendments after final rejection contained in the brief.

(5) Summary of Claimed Subject Matter

The examiner has no comment on the summary of claimed subject matter contained in the brief.

(6) Grounds of Rejection to be Reviewed on Appeal

The examiner has no comment on the appellant's statement of the grounds of rejection to be reviewed on appeal. Every ground of rejection set forth in the Office action from which the appeal is taken (as modified by any advisory actions) is being maintained by the examiner except for the grounds of rejection (if any) listed under the subheading "WITHDRAWN REJECTIONS." New grounds of rejection (if any) are provided under the subheading "NEW GROUNDS OF REJECTION."

(7) Claims Appendix

The examiner has no comment on the copy of the appealed claims contained in the Appendix to the appellant's brief.

(8) Evidence Relied Upon

5,774,661	CHATTERJEE et al.	6-1998
6,430,538	BACON et al.	8-2002
7,013,316	HANSEN et al.	3-2006
5,745,687	RANDELL	4-1998

Admitted prior art (as a result of untimely and/or improper challenge of Official Notice) that it is old and well known in the workflow management art to be able to indicate a task to be executed either synchronously or asynchronously.

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Official Notice

1. In the previous Office Action mailed March 6, 2008, notice was taken by the Examiner that certain subject matter is old and well known in the art. Per MPEP 2144.03(c), these statements are taken as admitted prior art because no traversal of this statement was made in the subsequent response. Specifically, it has been taken as prior art that:

- It is old and well known in the workflow management art to be able to indicate a task to be executed either synchronously or asynchronously.

Claim Rejections - 35 USC § 112

2. The following is a quotation of the first paragraph of 35 U.S.C. 112:

> The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

3. Claims 1, 3-6, 8-11, 13-16, 18-21, 23-26, 28-31, and 33-36 are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. The claim(s) contains subject matter which was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the inventor(s), at the time the application was filed, had possession of the claimed invention.

Claims 1, 11, 21, and 31 recite the step of providing each of multiple users the ability to specify a corresponding custom task associated with an extension point in a work flow "without editing said work flow". The specification does not provide written description or support for the exclusion of editing work flow.

Claims 306, 8-10, 13-16, 18-20, 23-26, 28-30, and 33-36 are dependent on claims 1, 11, 21 and 31, and thus are also rejected.

Claim Rejections - 35 USC § 103

4. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

5. Claims 1, 3, 4, 8-11, 13-14, 18-21, 23-24, 28-31 and 32-34 are rejected under 35 U.S.C. 103(a) as being unpatentable over **Chatterjee et al.** (US Patent # 5,774,661) in view of **Hansen et al.** (US Patent #7,013,316) and **Bacon et al.** (US Patent #6,430,538).

As per claim 1, Chatterjee et al. discloses a method of enabling a user to extend a work flow for synchronization/consolidation of data between at least two data sources, said work flow for execution in a meta directory server, said method comprising:

providing, by a designer, a sequence of built-in tasks which together when executed implement said work flow **(col. 3, lines 60-62, Workflow builder 214 is a subsystem that generates links or "maps" to define the steps, rules, and operations of a workflow) {a workflow is a sequence of steps and tasks}**, a built-in task in said sequence of built-in tasks containing an extension point at a point of interest in said work flow for users **(Figure 3 depicts a workflow built by a user and comprising a plurality of tasks; col. 5, lines 59-61; col. 6, lines 35-37; col. 7, lines 11-13 and 17-22; col. 13, lines 60-66; Default, built-in, or previously-defined tasks, processes, and operations are available for creating workflows. Existing workflows may be modified to create new ones. Decision point objects, or extension points, provide branching from one workflow to another.)**;

said work flow being designated for execution by multiple users as corresponding instances **(col. 3, lines 52-53, Workflow administrator 216 is a subsystem that sets up users; col. 6, lines 61-63, Distribution points allow several users to follow a single path in a workflow)**,

receiving from said user data indicating a custom task associated with said extension point wherein said custom task is separate from said sequence of built-in tasks **(col.5, lines 51-62, Workflow menu is used to set up , define, and verify new workflows. The choices in the workflow menu are New, Open, Import, Export, Define, Verify.. New is used to create a new workflow or operation, Open and Close are used to control access to existing workflows, Import and Export allow a user to store, remove, or add workflows; col. 6, lines 36-37, users are allowed to**

choose an operation for a step from a list of previously-defined operations) {built-in tasks are kept in a separate list, retrieving built-in tasks is a separate process than defining new tasks} and contains a program logic specified by said user for the corresponding desired customization **(col. 3, lines 60-62, col. 3, line 66-col. 4, line 2, col. 5, lines 58-59, col. 6, lines 9-26, 54-56; Creating or defining a new workflow, or the steps, rules and operations of a workflow, defining steps for a workflow, adding insertion points to a [preexisting] workflow diagram)**;

executing said custom task when said extension point is reached during execution of said built-in task in an instance of said work-flow for said user **(col. 7, line 62-col. 8, line 14; A decision point helps to execute conditional branching for a workflow; col. 6, lines 46-50, A Detour mode allows insertion of a "detour" path to or from a workbasket or operation that is temporarily unavailable.. The Flow Control menu provides for insertion of flow control points such as insertion, distribution, and decision; col. 7, lines 8-11, Workflow/Complex Operation, also accessible by button 358, permits insertion of a sub-workflow or complex operation into a current workflow diagram)**; and

continuing execution of said sequence of built-in tasks from said extension point in said built-in task after executing said custom task in said instance such that all of said sequence of built-in tasks are executed **(decision point 312 checks such data for certain properties, defined by the user in a manner described in connection with FIG. 4. As a result of this decision point 312, work flow processing will be routed either for further data entry 313 or to connection object "Accepted PO**

Path" 317. In the latter case, connection object "Accepted PO Path" 317 joins data from decision point 312 and decision point 314, database retrieval object 318 accesses a selected database and retrieves a specific record therefrom, and icons 319 and 320 indicate that the workflow is complete. Otherwise, data entry object 313 prompts the user for further input, decision point object 314 provides routing in response to the input data, and as a result the workflow either progresses to connection object Accepted PO Path317, indicating acceptance, or to work basket object 315, indicating rejection. In the latter event, work basket object 315 performs further processing, such as presenting a purchase order to a particular user as a rejected purchase order, at which point the work flow is complete, as indicated by icon 316; Rule engine 230 then evaluates 506 each condition, and a check 507 is made after each such evaluation to determine whether there are more conditions. When there are no more conditions for the current clause, a check 508 is made to determine whether there are more clauses… After all clauses have been evaluated, processing is done 509) {the workflow process is executed until no clauses or conditions remain} [Figures 3, 5, Column 7, line 64 – Column 8, line 14, Column 8, line 58—Column 9, line 3].

The workflow process discussed by Chatterjee et al. is not explicitly directed towards synchronizing/consolidating data between at least two data sources.

However, Hansen et al. teaches a sequence of tasks for synchronizing multiple databases stored on multiple computing devices, which include the steps of determining whether the databases are already initialized, verify the existence of a database, determine database configuration from a user, and update the server computer database as the user changes their data on the client computer, as well as query the server computer for any changes since the previous synchronization to determine what data is to be sent from the server computer to the client computer [abstract, Figures 2-6, col. 7, line 57—col. 8, line 18].

Both Hansen et al. and Chatterjee et al. are directed towards executing a sequential process of built-in subtasks for accomplishing a task; thus, they are reasonably pertinent to each other and are analogous references. Therefore, it would have been obvious to one of ordinary skill in the art at the time of invention to modify the teachings of Chatterjee et al. to be directed towards synchronizing/consolidating data between at least two data sources, because doing so ensures that a plurality of remotely located users are making decisions based on the same, consistent data, thereby reducing mistakes and errors in the decision-making process.

Although Chatterjee et al. teaches sequences of built-in tasks being used by many users and also that each user can provide custom extensions to said work flow by providing a corresponding instance of said program logic for said custom task **(col. 7, lines 8-19, Workflow/Complex Operation, also accessible by button 358, permits**

insertion of a sub-workflow or complex operation into a current workflow diagram. By defining subworkflows and complex operations, design of future workflow is simplified by re-use of such sub-workflows and complex operations… The Tools menu permits selection of commonly used operations for insertion in a workflow, based on previously-created definitions) {the sequence of tasks in a workflow or subworkflow may be stored and reused, presumably in other workflow processes and by other users of the enterprise}, Chatterjee does not explicitly teach the steps of:

providing each user with the ability to specify a corresponding custom task associated with said extension point without editing said work flow, the custom task specified by a user containing corresponding program logic to provide a customization desired by the user,

said flow being designed to execute the specified desired custom task in the corresponding instance if specified by corresponding user at said extension point;

However, Bacon et al. provides a workflow management system in which a defined sequence of personal subflow activities may be performed by one participant and instantiates and initializes a decision point agent with the branch expressions defined for the personal workflow [col. 9, lines 14-19, 52-59] (i.e., providing each user with the ability to specify a custom task associated with an extension point without editing the work flow, the custom task containing logic to be executed that provides the customization desired by the user). Personal subflows are stored so that it may be

used later in defining a workflow [col. 9, lines 44-45]. In step 710, work items are eventually routed to the participant who is defined within the workflow as the actor to perform the personal subflow for processing of the personal subflow activities (i.e., the custom task is executed in the corresponding instance as specified by the corresponding user at the extension point).

Both Chatterjee et al. and Bacon et al. are directed towards executing a defining and executing built-in tasks for accomplishing a workflow; thus, they are reasonably pertinent to each other and are analogous references. Therefore, it would have been obvious to one of ordinary skill in the art at the time of invention to modify the teachings of Chatterjee et al. to be directed towards providing users with the ability to specify a corresponding custom task associated with an extension point without editing a work flow that provides a customization desired by the user, the flow being designed to execute the specified desired custom task in the corresponding instance if specified by corresponding user at said extension point, because doing so improves the reusability of subprocess definitions and interoperability, as stated by Bacon et al. [col. 3, lines 23-35 and 31-33].

As per claim 3, Chatterjee et al. discloses wherein said custom task contains an another extension point, said method further comprises receiving from said user data indicating an another custom task to be executed when said another extension point is reached during execution of said custom task **(col. 7, lines 57-62, Execution of "data**

frm entry" object 311 provides the user with a prompt to enter data; col. 8, lines data entry object 313 prompts the user for further input, decision point object 314 provides routing in response to the input data, and as a result the workflow either progresses to connection object Accepted PO Path 317, indicating acceptance, or to work basket object 315, indicating rejection; col. 8, lines 50-67; Figures 3 and 5; A workflow can have more than one decision points for conditional branching.).

As per claim 4, Chatterjee et al. discloses further comprising:

determining a corresponding set of extension points available in each of said sequence of built-in tasks, displaying each of said set of extension points associated with a corresponding one of said sequence of built-in tasks, displaying said custom task and said another custom task and enabling said user to specify said custom task associated with said extension point, and said another custom task associated with said another extension point **(col. 8, lines 15-41; Figures 3-4; A workflow builder display allows a user to customize a workflow by inserting decision points, where the decision points come from a set of predefined conditional statements.)**.

As per claim 8, Chatterjee et al. discloses wherein at least one of said two data sources comprises a relational database **(item 318 in Figure 3)**.

As per claim 9, Chatterjee et al. discloses further comprising providing an utility to indicate that a specific one of said extension points is reached **(col. 8, line 50-col. 9, line 13; Figure 5; Conditional statements provide a check for additional conditions, clauses or other objects.)**.

As per claim 10, Chatterjee et al. discloses further comprising providing an utility in each of said sequence of built-in tasks and said custom task, wherein said utility indicates extension points available in a corresponding task **(col. 8, line 50-col. 9, line 13; Figure 5; Conditional statements provide a check for additional conditions, clauses or other objects.)**.

Claims 11, 13-14, 18-21, 23-24, 28-31 and 33-34 recite subject matter similar to that already rejected above.

Therefore, claims 11, 21, and 31 are rejected on the same basis as claim 1 above.

Claims 13, 23 and 33 are rejected on the same basis as claim 3, and claims 14, 24 and 34 are rejected on the same basis as claim 4.

Claims 18-20 and 28-30 are rejected on the same basis as claims 8-10 above.

6. Claims 5-6, 15-16, 25-26, and 35-36 are rejected under 35 U.S.C. 103(a) as being unpatentable over **Chatterjee et al.** (US Patent # 5,774,661), in view of **Hansen et al.** (US Patent #7,013,316) and **Bacon et al.** (US Patent #6,430,538), as applied to claims 1, 11, 21 and 31 above, and further in view of **Randell** (US Patent #5,745,687).

As per claim 5, Chatterjee et al. teaches enabling said user to specify that said custom task is to be executed synchronously, wherein said custom task is executed in a synchronous manner **(Workflow Stop, also accessible by button 352, inserts a stop point in a workflow)** [col. 6, lines 56-58].

Further, it has been **admitted as prior art**, as a result of untimely/improperly challenged Official Notice, that it is old and well known in the workflow management art to be able to indicate a task to be executed either synchronously or asynchronously. Therefore, at the time of the invention, it would have been obvious to a person of ordinary skill in the art to modify Chatterjee et al. to enable a user to specify a task to be executed synchronously or asynchronously as doing so provides the user with more control over when and how the task is executed, thereby enhancing the workflow design features offered in the workflow builder of Chatterjee et al.

Chatterjee et al. does not explicitly disclose that:

execution of said sequence of built-in tasks is resumed after completion of execution of said custom task such that said custom task is executed in a synchronous manner.

However, Randell teaches resuming a workflow sequence after completing execution of another task **(The node 314 could wait for all previous nodes to complete, or any combination to complete, before proceeding; An additional features contained in the Distributed Workflow system is manual coordination of a process to suspend the processing of an instance, resume processing of the instance from the point where it was suspended; The RESUME process allows a suspended instance to continue processing. Once RESUMEd, the coordinator will apply all the operations that were suspended while the instance was suspended, by placing the suspended nodes in the work queue)** [col. 8, lines 22-24, col. 16, lines 16-18, 57-63].

Both Chatterjee et al. and Randell are directed towards defining and executing workflow processes; thus, they are analogous references as they are directed toward a similar field of endeavor. Therefore, it would have been obvious to one of ordinary skill in the art at the time of invention to modify the teachings of Chatterjee et al. to include the step of synchronous execution of custom tasks and built-in tasks, because doing so would enhance the features in the Flow Control menu to permit insertion, distribution,

and stopping points of a workflow to execute custom tasks required as input for work flow processing, as contemplated by Chatterjee et al. [col. 6, line 49 – col. 8, line 14].

As per claim 6, Chatterjee et al. teaches enabling said user to specify that said custom task is to be executed asynchronously **(Distribution points allows several users to work on an item in parallel)** [col. 6, lines 61-62].

Further, it has been **admitted as prior art**, as a result of untimely/improperly challenged Official Notice, that it is old and well known in the workflow management art to be able to indicate a task to be executed either synchronously or asynchronously. Therefore, at the time of the invention, it would have been obvious to a person of ordinary skill in the art to modify Chatterjee et al. to enable a user to specify a task to be executed synchronously or asynchronously as doing so provides the user with more control over when and how the task is executed, thereby enhancing the workflow design features offered in the workflow builder of Chatterjee et al.

While Chatterjee et al. discusses asynchronous execution of a task, Chatterjee et al. does not explicitly disclose that multiple tasks are executed in an asynchronous manner.

However, Randell teaches multiple execution of tasks within a workflow by using performing the tasks in parallel **(Routing node 306 splits the specification created by**

work node 304 to allow three additional work nodes to perform parallel operations within the procedure) {a task is divided into multiple jobs that are executed in parallel} [col. 7, lines 46-48].

Both Chatterjee et al. and Randell are directed towards defining and executing workflow processes; thus, they are analogous references as they are directed toward a similar field of endeavor. Therefore, it would have been obvious to one of ordinary skill in the art at the time of invention to modify the teachings of Chatterjee et al. to include the step of asynchronous execution of custom tasks and built-in tasks, because doing so would enhance the ability of Chatterjee et al. to perform parallel execution of tasks at distribution points of the workflow, as contemplated [col. 6, lines 61-62]

Claims 15-16, 25-26, and 35-36 recite subject matter similar to that already rejected above. Therefore, claims 15-16, 25-26, and 35-36 are rejected on the same basis as claims 5-6 above.

(10) Response to Argument

With respect to the rejection of claims 1, 11, 21 and 31 raised under 35 USC 112, 1st paragraph, Appellant argues that the subject matter of the instant application is clearly in the predictable arts and that Figure 9 clearly demonstrates work flow 920 (containing built-in tasks 910-919), which is not edited, while any of a desired custom

task is associated with respective extension points The Appellant asserts that the claimed "not edited" feature is inherent to the disclosure of the subject application.

The Examiner respectfully disagrees. Figure 9 demonstrates that the sequence of Built-In Tasks 910-919 are not edited, but allows that each Built-In Task may correspond to an extension point, and each extension point might be associated with a custom task. The Examiner asserts that by allowing the extension point of a Built-In Task to be associated with any of a plurality of custom tasks constitutes constructing a customized, modified workflow sequence. Although the custom task itself or the sequence of Built-In Tasks may not have been edited or modified in any way, the Examiner asserts that the modularity of the custom tasks and how they may be interchangeably used with each extension point in each Built-In Task Examiner results in the work flow being edited. For this reason, the claimed "not edited" feature is ***not*** inherent to the disclosure of the instant application. For example, for Built-In Task 1, the extension point may be associated with Job 1, or Job 2, or Job 3, which is executed before the workflow returns to Built-In Task 2. The Examiner asserts that a workflow comprising of performing Job 1 within Built-In Task 1 is different than a workflow performing Job 3 within Built-In Task 1; the workflow has been modified to change the association of extension point of Built-In Task 1 from Job 1 to Job 3, therefore the workflow has indeed been edited. Further, the Examiner maintains the position that the specification does not describe in such a way as to reasonably convey to one skilled in the relevant art that the inventors at the time the application was filed, had possession

of the step of specifying a custom task associated with an extension point in a work flow "without editing said work flow".

Appellant argues that Chatterjee requires editing of the workflow, contrary to the recitation of claim 1.

The Examiner respectfully disagrees. Chatterjee allows a work flow sequence of tasks to be created and executed. The claimed method requires for a sequence of built-in tasks to be provided, wherein said sequence implements a workflow. Chatterjee provides users with the ability to create new workflows [Column 5, lines 51-61], as well as allowing users to assign specific operations, workbaskets and so on to a workflow step [Column 6, lines 22-26]. Chatterjee also allows insertion of a sub-workflow or complex operation into a current workflow diagram, simplifying design of future workflows by permitting re-use of such sub-workflows and complex operations [Column 7, lines 8-13]. Although Chatterjee provides users with the ability to modify existing workflows, Chatterjee also provides the ability to create a ***new*** workflow so that it has custom tasks or custom predefined sub-workflows associated with each workflow tasks where the custom tasks and predefined sub-workflows are modular components that may be used with a task within the workflow sequence of tasks, wherein the custom tasks and predefined sub-workflows have not been edited. The Examiner asserts that there is a difference between defining a new workflow and editing a workflow, namely that editing requires a workflow to have already been defined. In short, Chatterjee

allows editing of a workflow, but in no way whatsoever ***requires*** editing of a workflow as alleged by the Appellant. Thus, the Examiner asserts that a workflow in Chatterjee may simply have specific operations or workbaskets assigned to a workflow step, whilst the sequence of workflow tasks remains unedited.

Appellant argues that Bacon does not provide the ability to specify the personal subflow to be executed.

The Examiner respectfully disagrees. Bacon allows each user to instantiate and initialize a defined sequence of personal subflow activities to be performed. The entry point of the personal subflow may not be the starting point of a workflow [Column 9, lines 6-12]. In other words, the personal subflow must be attached or assigned to a corresponding activity within the workflow and not the starting point. The activity which should be next in the personal subflow is specified given a set of existing conditions [Column 9, lines 27-30]. The personal subflow is stored so that it can be used later in defining a workflow, and the workflow defined may include the personal subflow and assigned to a given participant, at which point the actor to which the personal subflow is assigned is then bound with the personal subflow [Column 9, lines 44-50]. When the server is initialized with a workflow definition that contains an activity to be performed by a personal subflow, the work item is eventually routed to the participant who is defined within the workflow as the actor to perform (i.e., execute) the personal subflow [Column 9, lines 52-63].

Appellant argues that Chatterjee does not continue execution after the same point in the built-in task at which execution of the custom task was started. Appellant argues that the decision point of Chatterjee merely provides for a choice of one of multiple paths to be executed, whereas specifying a custom task at an extension point in the claimed invention implies that control returns to the same point upon completion of the custom task.

The Examiner respectfully disagrees. Chatterjee teaches a "Return to" button that inserts a "return to" step that routes an object to a user who created that object [Column 7, lines 40-43, Figure 4]. Each of the buttons the user to insert or perform an action within a workflow, such as linking steps, deleting choices, insertion of a "detour" path, insertion of flow control points, adding insertion points, inserting objects, inserting stop points, inserting distribution points, collection points, broadcast points, workbaskets and sub-workflows or complex operations. As seen from Figure 3 of Chatterjee, execution of the work flow continues from decision point 312 (i.e., extension point) and may include processing of custom tasks/activities. From decision point 312, processing may continue to Accepted PO Path 317, or to Data Entry 313. From Data Entry 313, processing may continue to Decision Point 314. From Decision Point 314, processing may return to Accepted PO Path 317 or continue to Rejected PO 315. Thus, from each of the decision points (i.e., extension points), execution of the work flow continues from the decision point and proceeds to the next step of the work flow sequence. This is

deemed to be analogous to the claimed step of returning control to the same point of the work flow execution upon completion of the custom task because once control returns to the built-in task where the custom task is executed via the extension point, the workflow continues to the next step of the work flow sequence of built-in tasks as the task at the preceding built-in task has already been completed. In example, in a simple authorization work flow comprising the sequence of 'Fill Out Appropriate Form' and 'Obtain Approval', an extension point at the built-in task of 'Fill Out Appropriate Form' might include the custom task of 'Obtain Form X'. After the custom task of 'Obtain Form X' is completed, control returns to the work flow sequence, but since the built-in task of 'Fill Out Appropriate Form' has already been completed by virtue of completing the custom task at the extension point of said built-in task, control then returns to the next step in the work flow sequence, or 'Obtain Approval'. Although the steps immediately following decision points may be custom tasks/activities, this is not precluded by the claimed invention. Further, the teachings of Chatterjee are in agreement with the specification at page 12, lines 14-17, which states "it may be appreciated an extension points may be present any where in a custom (and also built-in) task, and a sequence of custom tasks may be executed in response. In addition, the execution of the built-in task continues after the completion of performance of the custom tasks associated with the extension point."

With respect to claim 4, Appellant argues that Chatterjee does not disclose or suggest determining available extension points; specifically, Appellant argues that the

predefined conditional statements of Chatterjee cannot be reasonably equated to the claimed available extension points.

The Examiner respectfully disagrees. The Examiner notes that the specification does not set forth any special definition for "extension point", therefore the broadest reasonable interpretation has been applied. As recited in the claims, each "extension point" is associated with a custom task and contained within a built-in task of a workflow sequence of built-in tasks. In other words, the "extension points" are similar to a decision node within a flowchart, where they are associated with tasks that initiate a decision, and are associated with tasks that are performed subsequent to the decision being made. Chatterjee allows users to define a plurality of rules using conditional statements that are evaluated at decision points (i.e., extension points), as well as providing a selection of routing choices that will be available as rule results [Column 9, lines 38-46]. In other words, based on the evaluation of the conditional statement rules, only certain routes/paths of the workflow are available for subsequent execution. As the purpose of the claimed extension points is to dictate the custom task to be associated with it within the workflow, the Examiner asserts that the decision points of Chatterjee are indeed an analogous concept.

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

/Peter Choi/

Examiner, Art Unit 3623

/Beth V. Boswell/

Supervisory Patent Examiner, Art Unit 3623

Conferees:

Beth V. Boswell /bvb/

Supervisory Patent Examiner, Art Unit 3623

Vincent Millin /bvb/ for

TC 3600 Appeals Conference Specialist